PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : G10L 15/08, 15/18

A1

(11) International Publication Number: WO 00/31725

(43) International Publication Date: 2 June 2000 (02.06.00)

(21) International Application Number: PCT/GB99/03812

(22) International Filing Date: 16 November 1999 (16.11.99)

(30) Priority Data:
98309650.4 25 November 1998 (25.11.98) EP

(71) Applicant *(for all designated States except US)*: ENTROPIC LTD. [GB/GB]; Compass House, 80–82 Newmarket Road, Cambridge CB5 8DZ (GB).

(72) Inventor; and

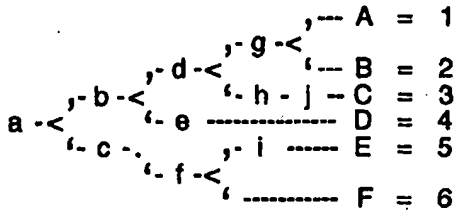
(75) Inventor/Applicant *(for US only)*: ODELL, Julian [GB/GB]; Entropic Ltd., Compass House, 80–82 Newmarket Road, Cambridge CB5 8DZ (GB).

(74) Agent: GILL JENNINGS & EVERY; Broadgate House, 7 Eldon Street, London EC2M 7LH (GB).

(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: NETWORK AND LANGUAGE MODELS FOR USE IN A SPEECH RECOGNITION SYSTEM



(57) Abstract

A language model structure for use in a speech recognition system employs a tree–structured network model. The language model is structured such that identifiers associated with each word and contained therein are arranged such that each node of the network model with which the language model is associated spans a continuous range of identifiers. A method of transferring tokens through a tree–structured network in a speech recognition process is also provided.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

NETWORK AND LANGUAGE MODELS FOR USE IN A SPEECH RECOGNITION SYSTEM

This invention relates to speech recognition systems and, in particular, to network models, language models and search methods for use in such systems.

One technique which is widely used in speech recognition systems is based upon the representation of speech units using probabilistic models known as hidden Markov models (HMMs). An HMM consists of a set of states connected by transitions. The HMMs are used to model units in the speech recogniser system which are usually individual speech sounds, referred to as phones. By using individual HMMs, models for complete words can be formed by connecting together individual phone models according to pronunciation rules for the language being recognised.

Given a segment of speech and a set of HMMs that may correspond to the speech segment, the likelihood that each set of HMMs corresponds to the speech segment can be calculated. If this likelihood is calculated for all words in a language vocabulary, then the most probable word can be chosen. One technique for doing this likelihood assessment employs the well-known Viterbi algorithm.

One approach to tackling the above problem has been to form a network model in which each word in a vocabulary is represented by a path through the model. Since the composite model is also an HMM, the most likely path, and hence the word, can be computed using the Viterbi algorithm. Such a model for single word recognition can be extended to the case of sentences by allowing connections from the end of words to the start of other words. So that language model probabilities, which are based upon the likelihood of one word being adjacent to another, can also be considered in such models, probabilities for each inter-word connections are also provided in such models.

Such network models can work well, but are often large and in order for them to be employed in real time require

considerable processing power. Furthermore, such models which only use a single HMM for each phone are often not particularly accurate. Accuracy can be improved by considering not only the identity of the phone the model represents but also the identity of the preceding and the following phone when determining the appropriate HMM parameters. Such an approach is often called a triphone approach. However, if the phonetic context is considered across word boundaries this approach increases the network complexity considerably. At word boundaries such a system requires that for each different cross-word boundary context a different HMM is used for the first and last phone of each word. This leads to considerable increase in network size and hence high computational requirements on the system employing such a model.

A number of approaches have been proposed in attempts to employ triphone models without excessive computational for those requirements. However, these approaches typically use approximate models and/or operate multiple passes through a network, reducing accuracy and/or increasing processing time.

The present invention seeks to provide a network model which can use such cross word context dependent triphone HMMs yet which overcomes the above and other problems.

As mentioned above, speech recognition systems usually require the calculation of likelihoods which must be computed to compare individual word hypotheses and determine the most likely word. If such a system employs word context as an assessment criteria, this usually means that such likelihoods are composed of two parts, the acoustic model likelihood (dependent upon the detected sound) and a language model probability. This language probability is normally determined from a reference language model which forms part of the speech recognition system, with the language model being accessed from the system network model as the network model is passed through during speech recognition. Given the large vocabulary and

high complexity of typical languages an accurate statistical model of general word sequences can be very large. The time taken to access this language model whilst carrying out the recognition process can be considerable, affecting significantly the system's ability to operate in real time, and the overall data processing requirement demands of the system.

The present invention seeks to provide a language model structure which stores all the necessary language model data, yet which is capable of being accessed quickly and efficiently.

According to a first aspect of the present invention, we provide, a language model structure for use in a speech recognition system employing a tree-structured network model, the language model being structured such that identifiers associated with each word and contained therein are arranged such that each node of the network model with which the language model is associated spans a continuous range of identifiers.

According to a second aspect of the present invention, we provide, a tree-structured network for use in a speech recognition system, the tree-structured network comprising:

a first tree-structured section representing the first phone of each word having two or more phones;

a second tree-structured section representing within word phones, wherein within word phones includes any phone between the first phone and the last phone of a word;

a third tree-structured section representing the last or only phone of each word;

a fourth tree-structured section representing inter-word silences; and,

a number of null nodes for joining each tree-structured section to the following tree-structured section.

Each tree structured section is joined to the next by a set of null nodes. These reduce the total number of links required and in the layer before the final phone

model of each word also mark the point at which the token history is updated to indicate the recognised word.

According to a third aspect of the present invention, we provide, a method of transferring tokens through a tree-structured network in a speech recognition process, each token including a likelihood which indicates the probability of a respective path through the network representing a respective word to be recognised, and wherein each token further includes a history of previously recognised words, the method comprising:

i) combining tokens at each state of the network to form a set of tokens, the set including a main token having the highest likelihood and one or more relative tokens;

ii) for each set of tokens, merging tokens having the same history;

iii) transferring the set of tokens to subsequent nodes in the network;

iv) updating the likelihood of at least the main token of each set of tokens; and,

v) repeating steps i) to iv) at each respective node.

Thus the present invention allows the tokens to be combined and then handled as sets of tokens. This helps reduce the amount of processing required to transfer the tokens through the tree-structured network..

According to a fourth aspect of the present invention, we provide, a method of merging sets of tokens in a speech recognition process, each token including a likelihood which indicates the probability of a respective path through the network representing a respective word to be recognised, and wherein each token further includes a history of previously recognised words, the method comprising:

i) assigning an identifier to each set of tokens, the identifier representing the word histories of each of the tokens in the set of tokens;

ii) comparing the identifiers of different sets of tokens; and,

iii) merging sets of tokens having the same identifiers.

The present invention allows identifiers to be assigned to sets of tokens, based on the histories of the tokens within the set. This allows different sets of tokens to be compared without requiring the comparison of the history of each token within each set, thereby reducing the level of computation required.

An example of the present invention will now be defined with reference to the accompanying drawings, in which:

Figure 1 is a schematic diagram showing an example HMM;

Figure 2 is a schematic diagram showing a prior art network model employing HMMs;

Figure 3 is a schematic diagram showing a known tree structured network;

Figure 4 is a block diagram of a speech recognition system employing the present invention;

Figures 5A, 5B and 5C are diagrams showing network fragments employed in a network model according to the present invention;

Figure 6 is a diagram showing a fragment of an example language model; and

Figures 7 is schematic diagram showing how a language model according to the present invention can be structured and accessed.

Referring to figure 1 the relationships between a received signal, observations based thereon, and an HMM network is shown. Its structure is similar to that described in the introduction above, with an entry node 1 and exit node n, with states 2...4 in between that, as a group, may represent a single phone or phones.

Figure 2 shows how a simple prior art network model is constructed from individual HMMs to produce individual

words. Included in the network are word end nodes 10 indicating that a word has been recognised.

Figure 3 shows a known tree structured network described in US Patent No. 5621859. In this model, within word triphone models are employed and a single tree used with approximate language model probabilities. With such an arrangement the exact language model (for a bigram) is unknown until the end of the word has been reached and depends upon the previous word. With such an arrangement it is possible to approximate within-word triphone nodes to reduce the computation required and forward searching can be complimented by backward searching which uses information from the forward pass and or/ acoustic models and/or language model. It has generally been considered, as this patent states, however, that full right and left context dependency is not possible with such a structure as it would create excessive branching and an exceptionally large network.

Figure 4 shows an example speech recognition system, with its key components, which can employ the network model 20 and language model 21 of the invention. The speech recognition system has means 22 for acquiring speech data and means 23 for encoding it. A recogniser 24 takes the encoded data and applies it, to the network model 20, which has access to a pronunciation dictionary 26, HMM information 27, and a language model 21. An answer 28 is then provided via a recogniser 24.

A network model 20 to represent all the words in the vocabulary is constructed that explicitly encodes the context-dependency of the acoustic models. This network includes connections from the end of each vocabulary word to the start of all legal following words (often all vocabulary words). The network model 20 is divided into three separate regions which are, where possible, tree-structured to share computation and are suitably connected to ensure that the context-dependent constraints are enforced. It should be noted that the language model

probabilities are not encoded in the network and a particular point in the network model 20 can form part of many words. Furthermore only at certain points in the network model 20 is the identity of the current word uniquely resolved.

The static network model 20 is built prior to decoding utterances and encodes information about the vocabulary items independent of language model information. The search procedure used in the decoder uses a modified time synchronous Viterbi algorithm with various pruning mechanisms to discard paths in the network for which the score (or likelihood) falls below a threshold. A path is a structure that holds information (the acoustic score, language model score and word history) to a particular point in the network. As each frame of speech is processed the paths are extended to connecting states. The network itself describes how paths can be extended. Since at any point in the network model 20 the word identity is in general unknown and for each possible word there can be multiple previous word sequences, it is necessary to store information about the various paths that can end at the same point in the network at a particular point in time. The head of these paths is encoded in a "token" structure which encodes the acoustic score to that point in the network up to the current time instant, the language model score and a pointer to previous words. These tokens are stored in the HMM state instance associated with active HMM. As the search proceeds new states are activated and token structures and other path-related information is created as needed.

Since the identity of the current word at an arbitrary point in the network model 20 is in general unknown, the language model probability that is applied to a path at any instant corresponds to the highest probability possible given the set of words for which the current network node forms part and the word history associated with the path. This language model probability can be refined as the

search approaches the word end points in the network model, at which point the word identity is fully resolved. As a consequence of this continual refinement the language model probabilities are used frequently in the search procedure and hence a cache is preferably used so that language model probability access can be performed efficiently. Furthermore an efficient language model structure to enable retrieval of language model probabilities in a tree-ordered structure forms a further aspect of the invention, and is described below.

The network structure will now be described.

The network model 20 is represented by a set of nodes and each node has an associated set of links which specifies the set of nodes which follow it in the network. Unlike many simple static networks there is no need to encode language model probabilities within the links as the language model is applied dynamically depending upon token histories.

In addition to unique initial and final nodes there are three types of nodes which occur in the network.

HMM nodes. These represent instances of particular HMMs. In addition to entry and exit states these contain instances of emitting HMM states at which token likelihoods are updated to reflect the likelihood of the associated state generating the acoustic observation.

Word-end nodes. These represent the points at which word identity becomes unique and are the points at which token history information is updated and token merging prior to the following word takes place. These nodes do not necessarily occur after the last phone of the word.

Null nodes. To minimise the number of links required between fully connected nodes, null nodes are added to the network. The addition of these collation points simplifies parts of the network which would otherwise require a large cross-bar of links.

For the rest of this discussion a general word is assumed to consist of a sequence of phones, eg:

WORD a b c .... x y z

The symbol a refers to a particular first phone which will be associated with a node of type A. Similarly the last phone of a word is z and the node associated with it as Z. In addition to these HMM nodes there are null nodes which occur between phones, together with word end nodes WE which are the points at which the token history is updated to reflect the recognition of WORD. The null nodes are identified by the two letters representing the position, ie. AB null nodes occur between the first and second phone models with ab representing a specific occurrence.

The recogniser uses context dependent triphone models in order to capture the coarticulation effects between adjacent phones and thereby improve performance. This produces more accurate models at the expense of increased HMM set size and a more complex recognition task.

In the following description, a triphone HMM for phone q preceded by phone p and followed by r will be referred to as p-q+r. However, it should be noted that although a separate logical triphone exists for each triphone context there are actually fewer distinct physical models due to sharing of parameters. In fact although there will be many tens of thousands of triphone contexts in a typical system the number of distinct HMMs is normally an order of magnitude smaller. In some of the diagrams this will be indicated by using the phone name followed by a number to indicate a model that may be shared in several different contexts.

For example a fragment of the network with the structure of figure 5a can be replaced with that of figure 5b if the model d1 was the one used for both triphones n-d+ae and n-d+b.

As a consequence of the use of triphone models, the example word above is represented by a sequence of logical triphone models:

WORD ?-a+b a-b+c b-c+d... w-x+y x-y+z y-z+?

It should be noted that when cross word context dependent models are used the identity of the model used for the first (and last) phone of the word is unknown until the identity of the previous (and next) word is known. Once the actual sequence of logical triphones is known, the sequence of actual modes needed to represent it can be found, eg

WORD a9 b36 c4 x8 y47 z30

In addition to basic triphone speech models this embodiment of the invention uses two context independent models for "silence". These are acoustic models encoding all of the various non-speech events and background noise that the recogniser should not transcribe as speech. A short pause, "sp", model is used in the periods where the speaker pauses briefly between words and the coarticulation effects of the previous phone are still present despite the pause. Such "sp" models are described as context free since they have no effect on context expansion.

Longer pauses can use a different silence model called "sil". This is used when the coarticulation effects are due not to the adjacent phone models but are due to the silence model. When expanding context "sil" models can be treated as any other phone apart from the fact that a context independent model is used for silence (with triphone models used for the actual phones). For example the phone sequence:

sil a b c sp d e sil f g h sil

would be expanded into the logical triphone sequence:

sil sil-a+b a-b+c b-c+d sp c-d+e d-e+sil sil sil-f+g f-g+h g-h+sil sil

In order to efficiently cope with cross word context dependent triphone acoustic HMMs the network model 20 is preferably built in four parts/stages. The core of the network model 20 is a simple left-to-right tree structured set of nodes representing phones 2 to (N-1) of all words in the vocabulary which have more than two phones. Each of these paths ends in a specific word end node which represents that point at which sharing stops and word identity becomes unique.

The first and last phone of the words are treated separately (as are words consisting of only one or two phones). All possible first phone models (the identity of which also depends on the last phone of the previous word) are arranged in their own separate section of the network. A network section is built for the last phone models (which depend upon the following word).

Finally the shared last phone network is linked to the shared first phone network with a network of inter-word silence models.

Interconnecting these three networks are four sets of null nodes which serve as collation points. As mentioned above, these are identified by the position they appear in the network: AB nodes occur between the first and second phone of the word, YZ nodes occur between the last two phones, ZS nodes occur between the last phone of the word and the inter-word silence and finally SA nodes occur between the inter-word silence and the first phone of the next word.

Thus, in general, each word has the following representation in the network.

SA a AB b.. y WE YZ z ZS s SA

With this structure it is obviously necessary to treat words which contain only a single phone, or just two phones differently. For two phone words the tree structured core of the network needs to be bypassed whilst special

provision is needed to cope with single phone words (which are replicated for different contexts and incorporated into the final phone layer of the network).

The collating null nodes are created for each context that actually occurs in the dictionary and all contextually consistent paths are joined with a shared network of HMMs. The set of YZ and AB contexts for which YZ and AB nodes are required can be found by scanning the dictionary. This scan can also be used to find the set of possible first and last phones. The product of these sets (together with "sil") defines the set of ZA contexts each requiring a ZS and an SA node. This network organisation is illustrated by Figure 5 which shows a network fragment corresponding to the words "and", "bill", "bit" and "but".

This network is constructed in the following stages.

i) Scan the dictionary to determine the set of contexts A, Z, AB and YZ.

ii) Make all required SA, AB, YZ and ZS nodes. The required set of ZS/SA nodes is the product of the A and Z sets (together with inter-word "sil" context).

iii) Join the YZ nodes to the ZS nodes using the appropriate HMMs (y-z+a) sharing these where possible.

iv) Join each ZS node to the appropriate SA node using the appropriate silence HMM ("sil" or "sp").

v) Join the SA nodes to the AB nodes using the appropriate HMMs (z-a+b) sharing these where possible.

vi) Scan the dictionary 26 by for each many phone word, find the appropriate AB node and, using models already present in the network where possible, add a path representing the phone sequence of the word ending in a word end node linked to the appropriate YZ node; for each two phone

word, add a word end node linking the appropriate AB and YZ nodes; and for each single phone word, for each possible preceding context(Z) create a word end node and link it to the appropriate SA node. Connect each of these word end nodes to the appropriate YZ nodes for all Z contexts.

Searching in the network model 20 and the efficient merging procedure used during the search will now be described.

The standard Viterbi algorithm stores a single token in each network state and updates this every frame by searching possible predecessor states and copying over the most likely. All other tokens are discarded.

In order to allow the generation of multiple hypotheses and to allow the later application of more detailed language models, the present invention is arranged so that alternative hypotheses are not discarded but a maximum, N, number of the most likely are retained.

If all likely hypotheses were retained and no simplification (or recombination) of hypotheses was allowed, the number of possibilities would grow exponentially with time. Fortunately as dependencies tend to be local rather than global, it is not necessary to separately process hypotheses which differ long into the past but share a common recent history. In other words the assignment of frames to states for a particular word may be dependent upon the immediately preceding word but is not affected by the choice of words further back in time. In practice when hypotheses share a common previous word they can be merged without degrading accuracy.

This means that rather than the select the best procedure employed for token propagation in the straightforward Viterbi algorithm the following more complex merging procedure is employed when multiple hypotheses are retained.

For each state of the network: Pass a copy of the token in each possible preceding state into the current state whilst updating its likelihood with the probability of the connecting transition and updating its path with required traceback information. Often it is not necessary to update the traceback information as the exact state/frame assignment is not needed only the most likely sequence of words. In this case the traceback information is only updated when a boundary is encountered.

Once all tokens have been collected, merge any sets of tokens which share the same previous word history (as it is assumed that they will follow the same path through the network so can share a single token with the variations represented by attaching multiple histories to the single token) and finally discard all but the N most likely.

This token merging operation is complex. However, it is found in practice that often the most likely N tokens all originate at the previous time instant from the same state and when this is true the merging procedure can be efficiently performed as follows.

Assuming that each set of tokens is stored as a main token, which represents the most likely hypothesis, together with a set of relative tokens which represent the alternative hypotheses. The relative tokens hold their likelihood not as an absolute number but as a likelihood relative to the main token of the set.

MAIN_TOKEN	likelihood	history
REL_TOKEN1	relative-likelihood	history
REL_TOKEN2	relative-likelihood	history
REL_TOKEN3	relative-likelihood	history

When token sets are stored this way, the complex merge operation can be bypassed when merging two token sets if all of the relative tokens of one set are identical to those of the other set. In this case all that is required is for the main token likelihood to be used to determine which token set is more likely. There is no need to examine each relative token in turn, determine that its history matches a relative token in the other set and decide which is more likely. Since all the pairs of relative likelihoods are the same, all the decisions will be made on the basis of the main token likelihoods. The relative tokens get propagated together with the main token.

If each token set is assigned an identifier that changes whenever the set of relative tokens changes, the determination of this condition can be made by comparing identifiers.

A further refinement is also possible. Although the language model scores will change as the token moves through the network, all tokens with the same history will change in the same fashion. This means that the token set identifier does not need to change due to the effects of changing language model likelihoods.

Overall the above procedure can significantly reduce the time taken to propagate tokens through the network during recognition.

In order to gain maximum benefit from these improvements and minimise search complexity, it is important not to delay application of the associated language model (or any other available knowledge source).

Although a single word identity is not known until the word end is reached, the set of possible words that each node of the network model 20 represents is known. If the network model 20 is viewed as a collapsed linear lexicon, the correct language model score for a token at a particular node is the maximum value over all words that the node represents.

In a direct implementation finding the maximum language model score for a particular node and history combination requires a number of language model probability look-ups equal to the number of words sharing the particular node (anywhere from 1 to a few thousand). Since each look-up requires significant computation, the computational cost of applying such an implementation at every point in the network is unacceptable and often the language model application would be delayed until the next word boundary.

However the invention reduces the complexity of the set of look-ups sufficient to allow continuous application of an NGram language model to become viable.

Figure 6 shows a typical way in which the language model may be stored. Each probability entry can be split up into the history, predicted word and a probability. When a backed-off or interpolated style of language model is used an additional weight is required for each history.

The complete language model can be efficiently stored and accessed using the following structures;

History entries: consisting of a history, a sparse array of probability entries and for some types of language model a weight.

Probability entries: consisting of a predicted word identifier and a probability.

There is a history entry for each history for which there are predicted words. The sparse array of probability entries hold the probability of the predicted word following the history. Note that the history can be any

length (including empty, in which case the probabilities are unigram probabilities).

Storing all the history entries in a hash table (hashed on the history) and sorting the sparse array according to the predicted word identifier enables a particular NGram probability to be found with a hash table lookup followed by a binary search of the sparse array.

This type of structure can also be used to enable fast look-up of the maximum LM likelihood at a particular network node. The identifiers associated with each word can be assigned in such a way that each node spans a contiguous range of identifiers finding the maximum likelihood over the set of words subsumed in a node is relatively simple. Rather than requiring a search for each word in turn, a single efficient binary can locate the first word for which there is an NGram entry and then the array can be processed linearly until the last word contained in the node is passed.

This ordering can be achieved for a left to right tree structured network by assigning predicted word identifiers according to their position in the tree (figure 7).

Although the whole network is not structured as a single left to right tree, each of the three parts is and so this numbering scheme can be used. However it should be noted that when a word has multiple pronunciations (ie a different phone sequence representing that word), it will have different routes through the network and thus may require different identifiers to ensure the contiguous assignment of word identifiers to nodes. This means that where previously the predicted word identifiers could be assigned one per word (or word class) they may now need to be different for different pronunciations. This increase in the size of the language model is usually small for word based language models, however, as most language model classes - i.e. words - only have one pronunciation.

Overall the above procedure reduces the cost of finding the maximum LM probability at each node by a factor

equal to the cost of the binary search (approximately O(log2(n)) where n is the number of probability entries in the particular history entry. In practice this is usually a factor in the range of 20-100.

CLAIMS

1. A language model structure for use in a speech recognition system employing a tree-structured network model, the language model being structured such that identifiers associated with each word and contained therein are arranged such that each node of the network model with which the language model is associated spans a continuous range of identifiers.

2. A tree-structured network for use in a speech recognition system, the tree-structured network comprising:

a first tree-structured section representing the first phone of each word having two or more phones;

a second tree-structured section representing within word phones, wherein within word phones includes any phone between the first phone and the last phone of a word;

a third tree-structured section representing the last or only phone of each word;

a fourth tree-structured section representing inter-word silences; and,

a number of null nodes for joining each tree-structured section to the following tree-structured section.

3. A method of transferring tokens through a tree-structured network in a speech recognition process, each token including a likelihood which indicates the probability of a respective path through the network representing a respective word to be recognised, and wherein each token further includes a history of previously recognised words, the method comprising the steps of:

i) combining tokens at each state of the network to form a set of tokens, the set including a main token having the highest likelihood and one or more relative tokens;

ii) for each set of tokens, merging tokens having the same history;

iii) transferring the set of tokens to subsequent nodes in the network;

iv) updating the likelihood of at least the main token of each set of tokens; and,

v) repeating steps i) to iv) at each respective node.

4. A method according to claim 3, wherein the likelihood of the relative tokens is a relative likelihood with respect to the likelihood of the respective main token.

5. A method of merging sets of tokens in a speech recognition process, each token including a likelihood which indicates the probability of a respective path through the network representing a respective word to be recognised, and wherein each token further includes a history of previously recognised words, the method comprising the steps of:

i) assigning an identifier to each set of tokens, the identifier representing the word histories of each of the tokens in the set of tokens;

ii) comparing the identifiers of different sets of tokens; and,

iii) merging sets of tokens having the same identifiers.

6. A method according to claim 5, wherein the step of combining the sets of tokens comprises the sub steps of:

a) selecting the main token with the highest likelihood;

b) forming a merged set of tokens with the selected token as the main token; and,

c) selecting from the remaining tokens a predetermined number of tokens having the

highest likelihood to form the relative tokens of the merged set of tokens.

7. A method according to claim 3 or claim 4, wherein the step of merging sets of tokens is performed in accordance with the method of claim 5 or claim 6.

8. A speech recognition system including a tree-structured network according to claim 2.

9. A speech recognition system including a tree-structured network, wherein the method of transferring tokens through the tree-structured network comprises a method according to claim 3 or claim 4.

10. A speech recognition system according to claim 9, wherein the tree-structured network is a tree-structured network according to claim 2.

11. A speech recognition system according to claim 9 or claim 10, wherein the method of merging sets of tokens comprises a method according to claim 5 or claim 6.

12. A speech recognition system including a language model according to claim 1.

13. A speech recognition system according to claim 12, when dependent on any of claims 8 to 11.

Fig.1.

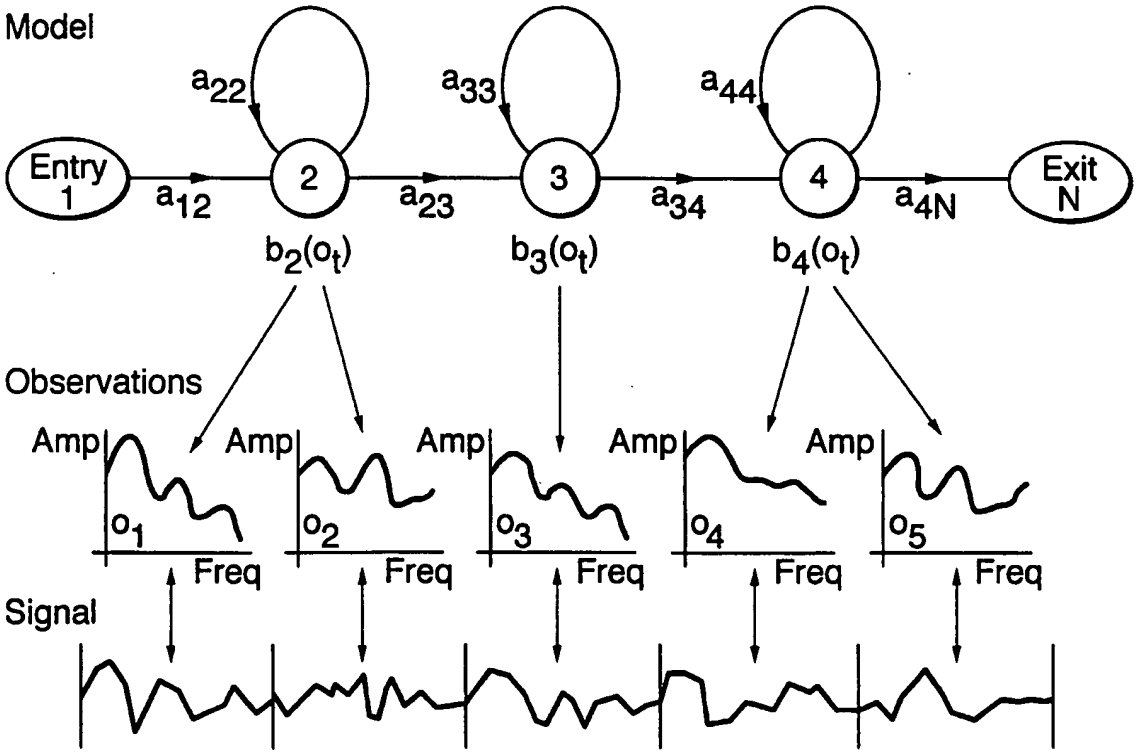

Model
a_{22}
a_{33}
a_{44}
Entry 1
a_{12}
2
a_{23}
3
a_{34}
4
a_{4N}
Exit N
$b_2(o_t)$
$b_3(o_t)$
$b_4(o_t)$
Observations
Amp
o_1
Freq
Amp
o_2
Freq
Amp
o_3
Freq
Amp
o_4
Freq
Amp
o_5
Freq
Signal

Fig.2.

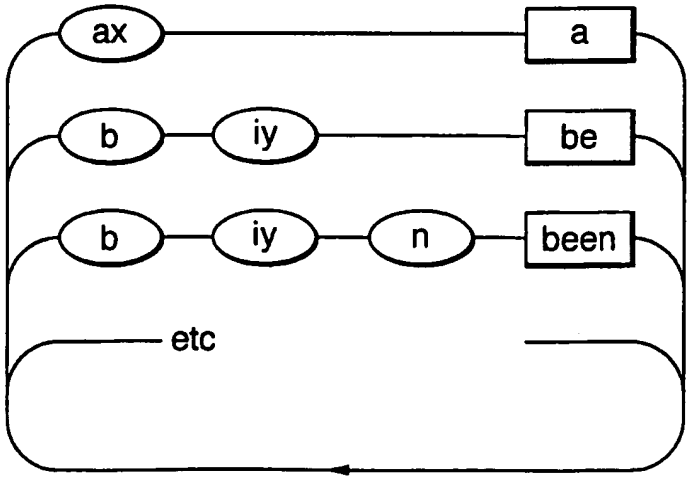

ax
a
b
iy
be
b
iy
n
been
etc

Fig.3.

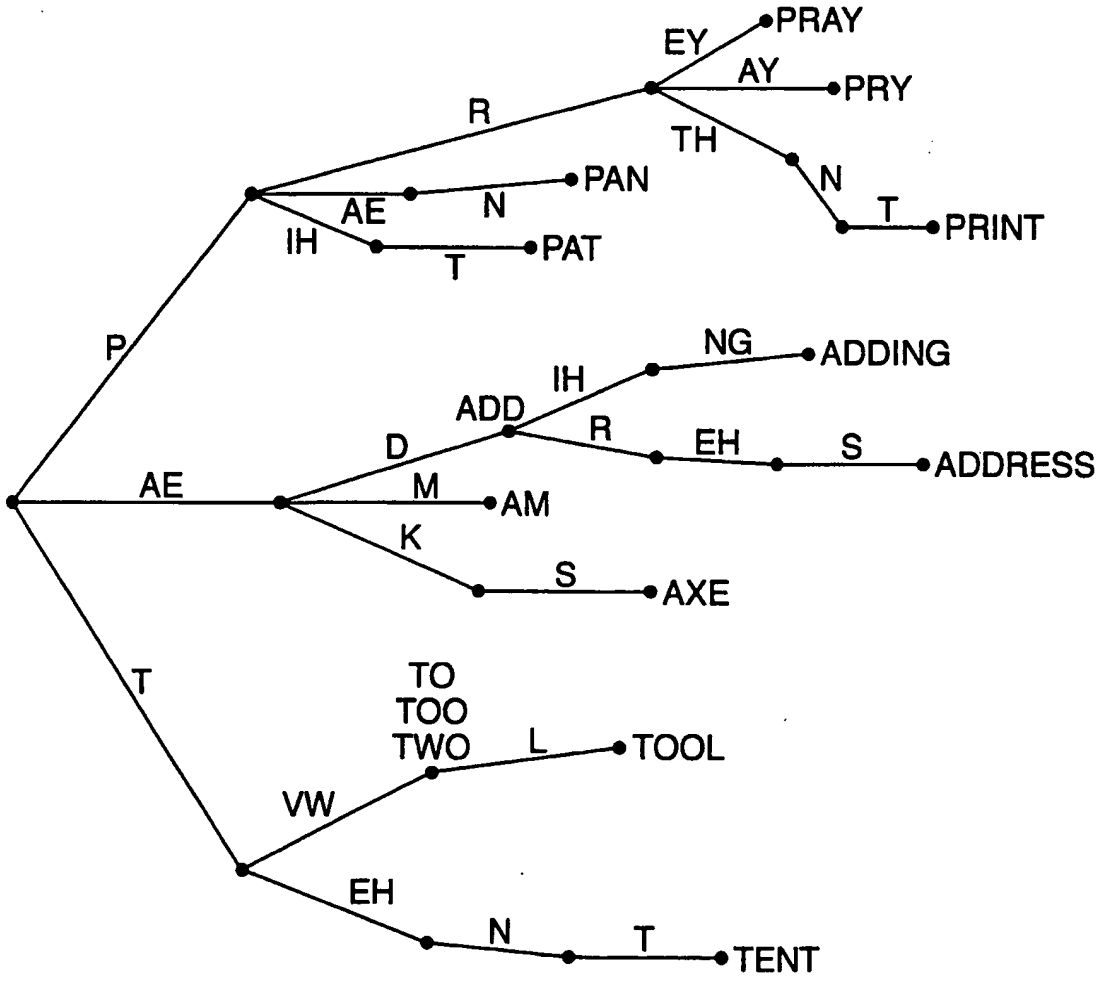

P
R
EY
PRAY
AY
PRY
TH
N
T
PRINT
AE
N
PAN
IH
T
PAT
AE
D
ADD
NG
ADDING
IH
R
EH
S
ADDRESS
M
AM
K
S
AXE
T
TO
TOO
TWO
L
TOOL
VW
EH
N
T
TENT

Fig.4.

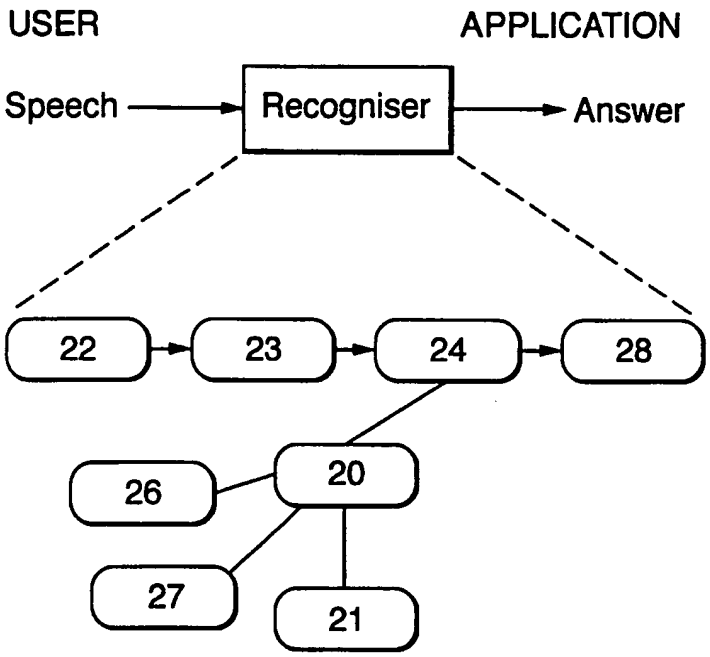


Fig.5a.

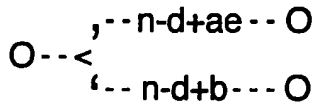


Fig.5b.

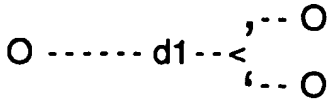
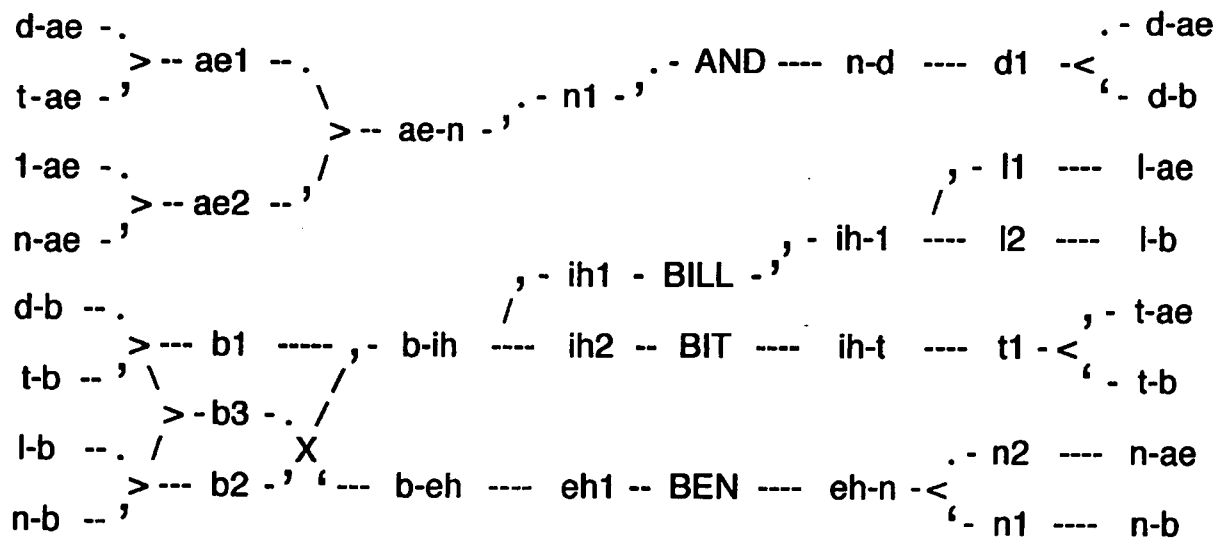


Fig.5c.

General Word Schema

ZA a AB b WE YZ z ZA

Network Fragment



Logical to Physical HMM Mapping

ae1 - d-ae+n t-ae+n
ae2 - l-ae+n n-ae+n
b1 - d-b+ih t-b+ih
b2 - l-b+ih n-b+ih
b3 - d-b-eh t-b+eh l-d+eh n-b+eh
d1 - n-d+ae n-d+b
eh1 - b-eh+n
ih1 - b-ih+1
ih2 - b-ih+t
l1 - ih-l+ae
l2 - ih-l+b
n1 - ae-n+d eh-n+b
n2 - eh-n+ae
t1 - ih-t+ae ih-t+b

Fig.6.

```
\data\
ngram 1=5
ngram 2=6
ngram 3=2

\1-grams:
-1.3275     </s>
-99.990     <s>                 -0.6336
-4.6878     APPLICATION         -0.5932
-4.7013     PATENT              -0.5672
-7.5000     ENTROPIC

\2-grams:
-4.7730     <s> </s>
-5.6668     <s> APPLICATION
-5.4180     <s> PATENT                  -0.2656
-1.0456     APPLICATION </s>
-1.3380     PATENT </s>
-1.8926     PATENT APPLICATION          -0.3456

\3-grams:
-0.0132     <s> PATENT APPLICATION
-2.0132     PATENT APPLICATION </s>
\end\
```

```
A   a b d g
B   a b d g
C   a b d h j   <-- Dictionary
D   a b e
E   a c f i
F   a c f
```

Fig.7a.

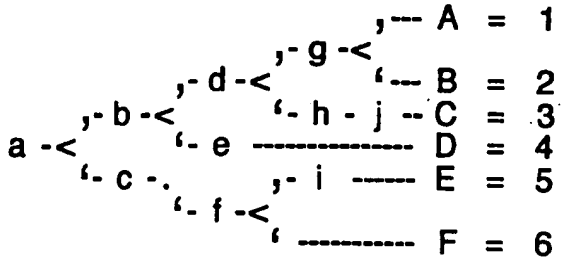


Fig.7b.

```
a = 1..6        f = 5..6
b = 1..4        g = 1..2
c = 5..6        h = 2
d = 1..3        j = 2
e = 4           i = 5
```

INTERNATIONAL SEARCH REPORT

International Application No
PCT/GB 99/03812

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G10L15/08 G10L15/18

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 732 685 A (ISTITUTO TRENTINO DI CULTURA) 18 September 1996 (1996-09-18) page 6, line 5 -page 8, line 5 ---	1
A	US 5 392 363 A (FUJISAKI TETSUNOSUKE ET AL) 21 February 1995 (1995-02-21) figure 28 column 18, line 63 -column 19, line 62 --- -/--	1

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
3 February 2000	10/02/2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Krembel, L

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/GB 99/03812

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	ORTMANNS S ET AL: "Language-model look-ahead for large vocabulary speech recognition" PROCEEDINGS ICSLP 96. FOURTH INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING (CAT. NO.96TH8206), PROCEEDING OF FOURTH INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING. ICSLP '96, PHILADELPHIA, PA, USA, 3-6 OCT. 1996, pages 2095-2098 vol.4, XP002099189 ISBN 0-7803-3555-4, 1996, New York, NY, USA, IEEE, USA paragraph '0003! ---	1
A	EP 0 720 147 A (AT & T CORP) 3 July 1996 (1996-07-03) figures 3A,3B,3C page 4, line 24 - line 41 ---	2,8
A	EP 0 238 692 A (IBM) 30 September 1987 (1987-09-30) page 2, line 45 - line 55 page 5, line 28 - line 50 page 20, line 21 - line 48 figure 26 ---	2,8
A	SCHWARTZ R ET AL: "A COMPARISON OF SEVERAL APPROXIMATE ALGORITHMS FOR FINDING MULTIPLE (N-BEST) SENTENCE HYPOTHESES" SPEECH PROCESSING 1, TORONTO, MAY 14 - 17, 1991, vol. 1, no. CONF. 16, 14 May 1991 (1991-05-14), pages 701-704, XP000245325 INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS ISBN: 0-7803-0003-3 * page 702, paragraph 2, "Lattice N-Best" * ---	3,5
A	LI Z ET AL: "New developments in the INRS continuous speech recognition system" PROCEEDINGS ICSLP 96. FOURTH INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING (CAT. NO.96TH8206), PROCEEDING OF FOURTH INTERNATIONAL CONFERENCE ON SPOKEN LANGUAGE PROCESSING. ICSLP '96, PHILADELPHIA, PA, USA, 3-6 OCT. 1996, pages 2-5 vol.1, XP002114710 1996, New York, NY, USA, IEEE, USA ISBN: 0-7803-3555-4 paragraph '02.2! -----	3,5

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/GB 99/03812

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0732685 A	18-09-1996	IT T0950200 A	17-09-1996
		US 5765133 A	09-06-1998
US 5392363 A	21-02-1995	JP 2667951 B	27-10-1997
		JP 7028949 A	31-01-1995
EP 0720147 A	03-07-1996	US 5805772 A	08-09-1998
		CA 2164458 A	01-07-1996
EP 0238692 A	30-09-1987	NONE	